**Assignment 1**

**Setup**

1. Create a new repo in GitHub called Assignment01.
2. Open the new repo / folder on your computer and proceed through the process of connecting Visual Studio Code to your remote repo.
3. Create a new HTML5 document and save it as index.html.
4. Add all of the necessary HTML5 code and title your document “Assignment 1”.
5. Create a new folder called js and then create an external JavaScript file titled script.js within the js folder and link it into the HTML document. You will be performing all of the tasks below within script.js.
6. Open script.js and add single-line comments for every step:  
   // STEP 1  
   // STEP 2  
   // STEP 3  
   etc.
7. Every step should be commented out when you push your final assignment to GitHub. So, you’ll do step 1, make sure you have it the way you want it to look or function, then comment it out. Then move to step 2. Make sure you have it the way you want it to look or function, comment it out, then move to step 3. Proceed like this until you’ve finished every step.

**Exercises**

1. Convert the following highlighted identifiers to Camel Case notation:  
     
   let some\_month;  
   function the Month();  
   let current-month;  
   let summer\_month;  
   let MyLibrary-function
2. Give me an example of a numeric literal expression, a string literal expression, a Boolean literal expression, and a null literal expression.
3. Give me two examples of complex / variable expressions.
4. Declare (but do not assign) 9 variables for the following identifiers: First Name, Last Name, Address, City, State, Zip Code, Your Age, Referral Source, May We Contact You. Use Camel Casing and Hungarian Notation when naming your identifiers.
5. Take the 3 of the 9 variables that you created above and demonstrate 3 ways for declaring and assigning values to those variables.
6. Create a variable.   
   Add a number and a string and display the coerced result in the browser’s console window.
7. Create two variables.   
   For the first variable, add a Boolean and a string and display the coerced result.   
   For the second variable, add a number and a Boolean and display the coerced result.
8. Is the following string literal valid? If not, how would you fix it?  
     
   let someString = 'Who once said, "Only two things are infinite, the universe and human stupidity, and I'm not sure about the former."';  
   window.console.log(someString);
9. Create a variable that produces a null value in the console window.   
   Then, create a variable that produces an undefined value in the console window.
10. Use the unary typeof operator on various literals to return the following types within the console window: string, number, Boolean, object, and undefined.
11. Within an alert box, use the concatenation operator (+) to display text in the alert box that appears as follows:   
      
    Hello Zak Ruvalcaba, welcome to the JavaScript class!  
      
    Substitute my name for your name. Although not necessary in practice, I want you to use 2 concatenation operators to construct this string of text. One after the text “Hello” and a second one after your name and before the comma.
12. Declare a variable called name and set it equal to your name.   
    Substitute your name in the previous alert string with the variable instead.
13. Declare a variable called course and set it equal to “JavaScript”.   
    Rework your alert string so that it displays the string of text but using the variables as opposed to hard coded text.
14. Rework the above string so that a line break is added right before the word “Welcome”. Your alert box should display as follows:  
      
    Hello Zak Ruvalcaba.  
    Welcome to the JavaScript class!
15. Replace the hardcoded string of your name with a prompt that asks the user for their name. The prompt’s response will now be captured in the name variable.
16. Replace the hardcoded string of the class you are taking with a prompt that asks the user for the class they are taking. The prompt’s response will now be captured in the course variable.
17. Declare a variable called x and assign it a value of 10.   
    Declare a variable called y and assign it a value of 20.   
    Display the sum of those two numbers in the console window.
18. Declare a variable called x and assign it a value of 20.   
    Add and assign 20 to that variable and display the result in the console window.  
    The result should be 40.
19. Declare a variable called x and assign it a value of 20.  
    Multiply and assign 5 to that variable and display the result in the console window.   
    The result should be 100.
20. Declare a variable called x and assign it a value that equals the remainder of 20 divided by 3.  
    Divide and assign 1 to that variable and display the result in the console window.   
    The result should be 2. If you got 6.66 try again.
21. Using a set of Comparison and Logical operators, write an application that evaluates to true and displays the result within the console window.
22. Using a set of Comparison and Logical operators, write an application that evaluates to false and displays the result within the console window. The application cannot use the same operators used in the previous application.